



Juho Suhonen

INFOTIEDOTTEET PROJEKTIHALLINTATYÖKALUSSA

INFOTIEDOTTEET PROJEKTIHALLINTATYÖKALUSSA

Juho Suhonen

Opinnäytetyö

Syksy 2011

Tietotekniikan koulutusohjelma

Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Koulutusohjelma	Opinnäytetyö	Sivuja	+	Liitteitä
Tietotekniikka	Insinöörityö	35	+	0
Suuntautumisvaihtoehto	Aika			
Ohjelmistojen kehitys	18.5.2011–30.8.2011			
Työn tilaaja	Työn tekijä			
Käyttösofta Oy	Juho Suhonen			
Työn nimi				
Infotiedotteet projektinhallintatyökalussa				
Avainsanat				
käytettävyys, web-pohjainen sovellus, PHP, MySQL, projektinhallinta				

Insinöörityössä suunniteltiin ja toteutettiin käyttäjäystävällinen web-pohjainen työkalun osio, jolla voidaan luoda helposti muistiinpanoja jo olemassa olevaan projektiin. Työn toimeksiantajana toimi Käyttösofta Oy, joka toimii Kempeleessä. Käyttösofta Oy kehittää asiakkailleen räätälöityjä web-pohjaisia sovelluksia asiakkaiden tarpeiden mukaan. Työn tarve syntyi yrityksen sisällä ja pääasiallisia käyttäjiä ovat Käyttösofta Oy:n henkilöstö. Työssä tuli ottaa huomioon käyttämisen helppous ja sovelluksen dynaamisuus.

Projekti-infon toteutus muistuttaa foorumipohjaista esitystapaa, jossa käyttäjät voivat käydä keskustelua kyseisestä projektista. Käyttäjät voivat kommentoida alkuperäistä infotiedotetta ja muokata omia tekstejään. Infotiedotteista selviää käyttäjän nimi, viestin aihe, viestin päiväys ja itse viesti. Jos sovellukseen lisätään uusi projekti, lisätään Luo projekti-info -toiminnon kautta uusi infotiedote, jossa käyttäjä luo alkutiedotteen uudelle projekti-infolle.

Sovellus on yhteydessä yrityksen MySQL-tietokantaan, johon tarvittavat tiedot tallennetaan ja josta ne haetaan käyttöön. Toteutuksessa käytettiin pääsääntöisesti PHP-, HTML/XML- ja CSS-kieltä. Toiminnallisuus web-sovellukseen luotiin käyttäen JavaScript-kieltä. Tuloksena työstä saatiin toimiva kokonaisuus, jolla voidaan käydä vuoropuhelua projekteista viestien avulla.

ABSTRACT

Degree programme	Thesis	Number of pages	+	appendices
Information Technology	Bachelors thesis	35	+	0
Line	Date			
Software Developing	18.5.2011–30.8.2011			
Commissioned by	Author			
Käyttösofta Oy	Juho Suhonen			
Thesis title				
Project information messages in Web-based software				
Keywords				
user-friendly, web-based software, PHP, MySQL, project management				

The main purpose of this thesis work was to develop a user-friendly management program tool for work projects, which handles project information messages. The work was made for Käyttösofta LLC, which operates in Kempele. Käyttösofta LLC creates user-friendly web-based software based on the demands customers. The topic for this thesis was created based on the needs of the company and the main user group for this software is the members of its staff. Main focus was set on two key features when developing the software; user-friendliness and dynamism.

The work consists of features such as creating new project-information, handling project information and making new notes of the chosen project. The project notes-feature resembles a forum type user interface in which users can discuss about on-going projects. The user can make comments about notes or edit their own comments. Project notes-feature shows the user user-information, date and project information notes. When a new project is added into the tool, the user is able to create new project info in which the user creates first information message regarding project info.

The software is connected to the company's database, where certain information is saved and brought up when needed. In this thesis, the following programming languages were used: PHP, HTML, XML, CSS. Also MySQL-language was used for databases and JavaScript-language for functionality of the pages. As a result of this thesis, a functional part of the software was developed in which the user can discuss about projects in the form of messages.

ALKULAUSE

Opinnäytetyönä toteutettiin viestittämistyökalu projekteille. Viestittämisosio luotiin Käyttösofta Oy:lle, joka toimii Kempeleessä. Osio on osa työkalua, jossa käyttäjä voi hallita vireillä olevia projekteja ja niihin liittyviä asioita. Työ tehtiin 18.5.2011–30.8.2011 pääasiassa Kempeleessä Käyttösofta Oy:n tiloissa.

Viestittämisosion tarve tuli Käyttösofta Oy:n puolelta, jossa toivottiin työkalua, joka helpottaisi viestittämistä projekteista työntekijöiden välillä. Toukokuun 2011 aikana pidettiin palaveri, jossa päätettiin aloittaa työkalun kehittäminen.

Yrityksen ohjaavana tahona toimi projektien vetäjä Jani Räihä. Räihän kanssa sovittiin päälinjat, joita noudatettiin työssä. Työtä ideoimassa oli myös mukana ohjelmistosuunnittelija Marko Sarala. Työtilat ja työkalut sain käyttöön yritykseltä. Haluaisinkin kiittää Käyttösofta Oy:n henkilöstöä onnistuneesta projektista ja kuluneesta kesästä.

Oulussa 31.8.2011

Juho Suhonen

SISÄLLYS

TIIVISTELMÄ	3
ABSTRACT	4
ALKULAUSE	5
SISÄLLYS	6
SANASTO	7
1 JOHDANTO	8
2 KÄYTETYT TEKNIIKAT JA SOVELLUKSET	9
2.1 PHP	9
2.2 MySQL	11
2.3 HTML	12
2.4 JavaScript	13
2.5 CSS	14
2.6 Kehitysympäristö	15
2.7 Dynaamiset ja staattiset web-sovellukset	16
2.8 Projektinhallintatyökalut	16
3 PROJEKTI-INFOTYÖKALUN TAULUT JA TOTEUTUS	18
3.1 Tietokannat	18
3.2 Työkalussa käytetyt taulut	19
4 PROJEKTI-INFOT	22
4.1 Työn vaatimukset	22
4.2 Sovelluksen käyttö	22
4.3 Toiminnot	24
4.4 Rakenne	28
4.5 Käyttöliittymä	29
5 YHTEENVETO	33
LÄHDELUETTELO	34

SANASTO

CSS (Cascading Style Sheets, porrastetut tyyliarkit) on erityisesti WWW-dokumenteille kehitetty tyyliohjeiden laji.

DBMS eli Database Management System on tietokannan hallintajärjestelmä.

HTML (lyhenne sanoista HyperText Markup Language) on ohjelmointikieli, jolla voidaan kuvata hyperlinkkejä sisältävää tekstiä eli hypertextiä. HTML:llä voidaan myös merkitä tekstin rakenne eli esimerkiksi, mikä osa tekstistä on otsikkoa ja mikä leipätekstiä. HTML tunnetaan erityisesti kielenä, jolla web-sivut on koodattu.

JavaScript on alun perin Netscape Communications Corporationin kehittämä pääasiassa Web-ympäristössä käytettävä komentosarjakieli. JavaScriptin tärkein sovellus on mahdollisuus lisätä web-sivuille dynaamista toiminnallisuutta.

Localhost on paikallinen palvelin.

MySQL on SQL-tietokannan hallintajärjestelmä.

PHP (lyhenne sanoista HyperText Preprocessor) on ohjelmointikieli, jota erityisesti käytetään web-pohjaisissa dynaamisissa web-sovelluksissa.

WAMP Server on ohjelmakokonaisuus, joka sisältää Apache-palvelimen, PHP:n ja MySQL:n. Wamp Server on tarkoitettu Windows-käyttöjärjestelmille.

XML (lyhenne sanoista Extensible Markup Language) on merkintäkieli tai standardi, jolla tiedon merkitys on kuvattavissa tiedon sekaan.

1 JOHDANTO

Työn tarkoituksena oli rakentaa Käyttösofta Oy:lle helppokäyttöinen ja dynaaminen työkaluosio. Osion päämääränä on helpottaa kommunikaatiota projektiryhmän välillä ja nopeuttaa projektin etenemistä. Käyttäjär ryhmänä työssä toimii pääasiassa Käyttösofta Oy:n henkilöstö.

Projektin alkuvaiheessa päätin luoda Luo projekti -infotoiminnon ensin, koska työkalussa oli vastaavanlainen toiminto projektin luonnille. Idea on, että käyttäjän tulee ensin luoda projekti-info jo olemassa olevasta projektista. Projekti-infot työssä löytyvät päävalikosta, josta voi valita haluamansa projekti-infon.

Viestittämisosio muistuttaa foorumityyppistä näkymää. Tässä osiossa käyttäjä voi helposti kommunikoida muiden työntekijöiden kanssa kyseisestä projektista. Käyttäjä voi myös muokata omia kommenttejaan tai luoda uuden ala-aihealueen. Käyttäjällä on mahdollisuus muokata kaikkien kommentteja, mikä ehkäisee turhien viestien luomisen, kun kyseessä on vain pieni lisäys viestiin. Infotiedotteista selviää käyttäjän nimi, kommentin aika ja aihe. Kommentteja voidaan poistaa tai merkata tehdyksi. Kommentit mielletään tehtäviksi, joten ne voi merkata tehdyiksi. Käyttäjä erottaa tehdyt ja tekemättömät tehtävät eri fonttiväristä.

Info-osio on yhteydessä yrityksen tietokantapalvelimelle, josta ne tiedot haetaan ja tallennetaan. Taulujen tarkemmat kuvaukset määritellään luvussa 4.1.

2 KÄYTETYT TEKNIIKAT JA SOVELLUKSET

2.1 PHP

PHP on komentosarjakieli, jossa ohjelmakoodi tulkitaan vasta ohjelman suoritussvaiheessa. PHP:tä voidaan käyttää useilla eri alustoilla ja käyttöjärjestelmillä. PHP:n ensimmäinen versio julkaistiin vuonna 1995, ja nykyisin PHP on vertailuissa johtava dynaamisten web-palveluiden tuottamiseen tarkoitettu kieli. (1.)

Alunperin PHP-lyhenne tuli sanoista ”Personal Home Page Tools” vuonna 1994, kun Rasmus Lerdorf aloitti PHP:n kehitystyön. Kahden version jälkeen siirryttiin käyttämään lyhennettä PHP:HyperText PreProcessor. Ensisijaisesti PHP:tä pidetään työvälineenä dynaamisten web-dokumenttien luomiseen ja se on erityisesti suunniteltu web-sovelluskehittämistä varten. PHP tarjoaa ammattilaisille nopean tavan kehittää web-sovelluksia ja vasta-alkajille matalan aloituskynnyksen. (2, s. 9–10.)

PHP:n syntaksi on suurimmaksi osaksi lainattu C-kielestä. PHP:stä löytyy myös piirteitä Java-, Perl- ja C++-kielistä. PHP:n avulla voidaan käyttää useita tietokantoja, kuten Oracle, IBM DB2, Ingres, MySQL ja Solid. (2, s. 9–10.)

Kuvassa 1 on esimerkki PHP:n syntaksista, jossa ruudulle tulostetaan tervehdysteksti.

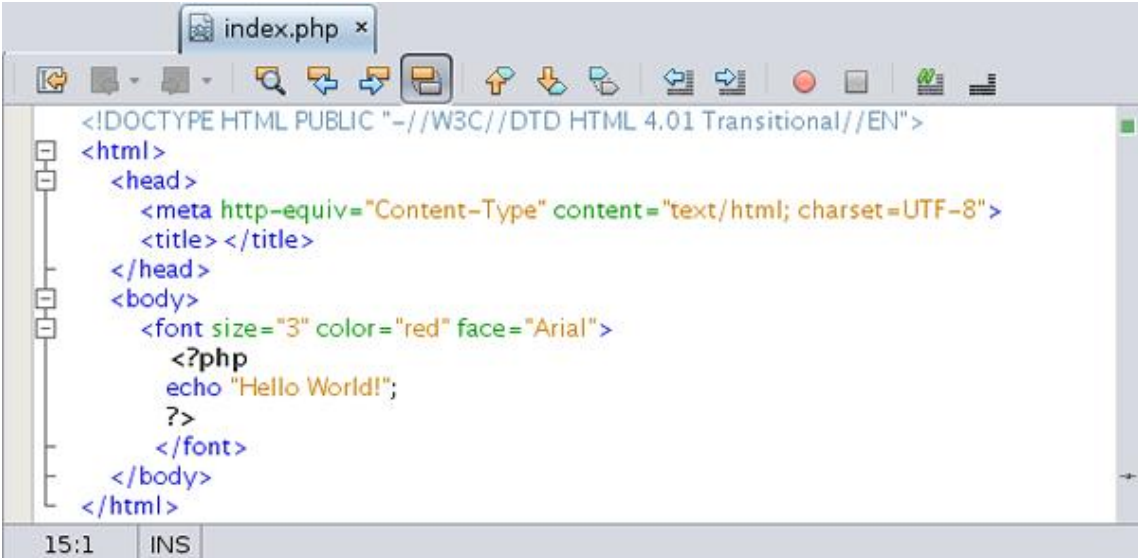
PHP	<code>echo "<h1>Hello, {\$_POST['name']}</h1>";</code>
XHP	<code>echo <h1>Hello, {\$_POST['name']}</h1>;</code>

KUVA 1. Esimerkki PHP-syntaksista

PHP on ilmainen, palvelinpohjainen skriptikieli, jota käytetään usein toiminnallisten www-sivujen toteuttamiseen. Palvelinpohjaisuus tarkoittaa sitä, että PHP-koodi suoritetaan palvelimella, jolloin se ei vaadi mitään erityistä tukea selaimelta. Se tarkoittaa myös sitä, että skriptillä on pääsy esimerkiksi palvelimen tiedostoihin ja tietokantoihin. (3.)

PHP on niin sanottu open source -ohjelmisto, mikä tarkoittaa, että ohjelmisto on avoin kaikille ja sen lähdekoodi on vapaasti saatavilla. Tyypillisesti PHP:tä käytetään Apache-web-palvelimen kanssa, esimerkiksi LAMP- tai Wamp Server -alustalla. (2, s. 9–10.)

Kuvassa 2 on HTML-koodiin sisälle on upotettu PHP-koodia.



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title></title>
  </head>
  <body>
    <font size="3" color="red" face="Arial">
      <?php
        echo "Hello World!";
      ?>
    </font>
  </body>
</html>
```

The image shows a code editor window titled 'index.php'. The code is an HTML document with a PHP script embedded in the body. The PHP code uses the 'echo' function to output the text 'Hello World!'. The HTML structure includes a DOCTYPE declaration, a head section with a meta tag for content type and charset, and a body section with a font tag that sets the font size to 3, color to red, and face to Arial. The PHP code is placed inside the font tag's content area.

KUVA 2. PHP-koodia

2.2 MySQL

MySQL on SQL-tietokannan hallintajärjestelmä, joka on asennettu yli kuuteen miljoonaan tietokoneeseen. MySQL:ää kehittää ruotsalainen yritys MySQL AB. MySQL-tietokanta on hyvin suosittu web-palveluiden tietokantana. MySQL-tietokannan päälle rakennettava ohjelmalogiikka tehdään usein PHP-, Python- tai Perl-ohjelmointikielellä. Sivut julkaistaan Apache-web-palvelimella, joka edelleen toimii Linux-käyttöjärjestelmän päällä. Tämän kaltaista alustaa kutsutaan LAMP-alustaksi. (4.)

Vuonna 1979 Michael Widenius loi MySQL:n edeltäjän UNIREG-tietokantatyökalun. Alun perin UNIREG oli tarkoitettu suppeampien tietovarastojen käyttöön. Myöhemmin työkalua on laajennettu tukemaan isompia tietovarastoja ja monille eri kielille. Vuonna 1994 ruotsalainen yritys nimeltä TcX ryhtyi kehittämään web-pohjaisia sovelluksia käyttäen UNIREG-työkalua. TcX huomasi, että UNIREGin avulla ei pystytty luomaan onnistuneita dynaamisia web-sivustoja, jonka seurauksena se vaihtoi mSQL-tietokantajärjestelmään. Vuonna 1995 yhteistyö Michael Wideniuksen ja mSQL:n luoja David Hughesin välillä tuotti MySQL version 3.11. (5, s. 5–6.)

SQL-kieli on luotu noudattamaan tarkkoja sääntöjä. Tietokoneet pystyvät kääntämään SQL-kyselyn nopeasti. SQL-kysely on täysin määritelty komento, joka lähetetään tietokantapalvelimelle, joka suorittaa kyseisen komennon. Kuvassa 4 on kuvattu esimerkki SQL-kyselystä, jossa haetaan people-nimisestä taulusta, name-nimisestä sarakkeesta kaikkia tuloksia, joissa on merkittynä Stac-sana. (5, s. 23.)

```
SELECT name FROM people WHERE name LIKE 'Stac%'
```

KUVA 4. SQL-kysely

2.3 HTML

HTML (Hypertext Markup Language) on standardoitu merkkaukieli web-sivujen tekoon. HTML:llä voidaan merkitä tekstin rakenne eli esimerkiksi, mikä osa tekstistä on otsikkoa ja mikä leipätekstiä. Tammikuussa 2008 julkaistiin HTML 5.0, joka on uusin versio. (6.)

HTML syntyi vuonna 1989, kun Tim Berner-Lee ja Robert Cailau suunnittelivat CERN (European Particle Physics Laboratory) -dokumenttien tilalle korviketta. HTML-kieli helpotti suunnittelijoita, kun enää ei tarvinnut koota kuva-, ääni- tai tekstikokoelmia. Merkittävämpänä parannuksena voidaan pitää hypertekstilinkkejä, jotka mahdollistivat dokumenttien linkittämisen toisiin dokumentteihin, vaikka niiden fyysinen sijainti olisikin eri puolella maailmaa. Kuvassa 5 on esimerkki HTML-koodista. (7, s. 3.)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML
<html>
  <head>
    <title>Example</title>
    <link href="screen.css" rel="sty
  </head>
  <body>
    <h1>
      <a href="/">Header</a>
    </h1>
    <ul id="nav">
      <li>
        <a href="one/">One</a>
      </li>
      <li>
        <a href="two/">Two</a>
      </li>
```

KUVA 5. HTML-koodia. (6.)

Perusrakenne HTML-koodissa merkitään HTML-tagien sisään. Jokainen tagi aukaistaan ja suljetaan. Jokaisessa HTML-dokumentissa on head- ja body-osio. Head-osio käsittelee dokumentin otsikkoa ja sitä, mitä muita tiedostoja halutaan sisällyttää kyseiseen HTML-dokumenttiin. Body-osioon luodaan sivun varsinainen sisältö, esimerkiksi kuvat ja tekstit.

(7, s. 37–38, s. 47–52).

HTML-dokumenttiin voidaan upottaa useita kieliä, kuten esimerkiksi PHP:tä, mikä mahdollistaa dynaamisten sivujen luomisen. Usein HTML-dokumentteihin voidaan sisällyttää JavaScript-tiedostoja tai upottaa suoraan HTML-koodin sekaan sivujen toiminnallisuuden parantamiseksi.

2.4 JavaScript

JavaScript on skriptikieli selainohjelmointiin. JavaScriptillä luodaan toiminnallisuutta verkkosivuille. Vuonna 1995 Navigator 2 -selaimen mukana julkaistu JavaScript 1.0 -kieli mahdollisti ensi kertaa interaktiivisen HTML-dokumenttien luomisen. JavaScript on tulkettava kieli, kuten muutkin skriptikielet, mikä edellyttää koodin kääntämisen ennen suorittamista. JavaScript-tulkki voidaan upottaa mihin tahansa WWW-selaimeen tai WWW-palvelimeen. Tulkattaessa JavaScript-ohjelmaa se suoritetaan rivi riviltä sitä mukaa kuin skriptiä luetaan. Etuna voidaan pitää JavaScript-ohjelman muokattavuutta ja sitä, että muutokset saadaan näkyviin päivitettäessä selainta. (8, s. 2–9.)

JavaScript soveltuu parhaiten lyhyiden ja yksinkertaisten ohjelmien tekoon. JavaScript ohjelmat voidaan kirjoittaa nopeasti ja sisällyttää suoraan HTML-koodiin. JavaScript-kielen syntaksi pohjautuu osittain Javaan, mutta kielillä on silti suhteellisen vähän yhtäläisyyksiä. Kuvassa 6 on esimerkki JavaScript-kielen käytöstä HTML-dokumentissa. (8, s. 9–10.)

```
<html>
<head>
<script type="text/javascript">
<!--
function PrintDate() {
    today = new Date();
    document.write('Date: ', today.getMonth()+1, '/', today.getDate(), '/', today.getYear());
}
//-->
</script>
</head>

<body>
<p align="right">
<script type="text/javascript">
<!--
PrintDate();
//-->
</script>
</p>
THE REST OF YOUR PAGE.
</body>
</html>
```

KUVA 6. Esimerkki JavaScript-koodista

JavaScript-koodia pystyy kirjoittamaan useilla ASCII-tekstieditoreilla tai HTML-editorilla. Osa editoreista kykenee tuottamaan automaattisesti koodia, kun taas osa HTML-editoreista värjää koodin avainsanat ja helpottaa tekijän työtä. JavaScript-koodi kirjoitetaan script-tagien sisälle, jotka useimmiten sijoitetaan HTML-koodissa head-tagin sisälle. Koodi kuitenkin voidaan sijoittaa myös body-tagien väliin. (8, s. 12–17.)

2.5 CSS

CSS on erityisesti WWW-dokumenteille kehitetty tyyliohjeiden laji. Tyyliohjeet määrittävät minkä näköiseksi web-sivusto tehdään. Tyyliohjeissa määritellään mm. fontit, värit ja muodot tai varjostukset. Tyyliohjeita tukevat useimmat julkaisuohjelmat kuten myös yleisimmät tekstinmuokkausohjelmat. (7, s. 234–235.)

CSS:ssä dokumentille voi määritellä useita tyyliohjeita, jotka yhdistetään tietyllä tavalla yhdeksi säännöstöksi. CSS-koodi voidaan kirjoittaa suoraan HTML-koodin sisään, mutta tyypillisempää on, että koodi sijoitetaan omaan tiedostoon. Kun luodaan oma css-tiedosto, sitä kutsutaan ulkoiseksi tyyliksi (external style). Ulkoinen tyylitiedosto on perus-ASCII-tiedosto, jossa tyylit on määritelty sääntöjen avulla. Haluttaessa käyttää ulkoista tyyliä sivustolla tulee kutsua tyylitiedosto head-tagien sisällä HTML-dokumentissa. Jos CSS-koodi kuitenkin sijoitetaan HTML-dokumenttiin, tulee se sijoittaa head-tagien sisälle ja käyttää style-tagia, kuten kuvassa 8. (9.)

```
<html>
<head>
<style type="text/css">
body {color:red;}
h1 {color:#00ff00;}
p.ex {color:rgb(0,0,255);}
</style>
</head>
```

KUVA 8. Tyylit sijoitettuna samaan tiedostoon

Kuvassa 9 on määritelty tekstikentälle mm. seuraavia tyylejä: tekstikentän reunoihin 1 pikselin levyinen jatkuva rajausta mustalla värillä, tekstikentän taustaväri, fontti ja fontin koko, reunojen pyöristys ja varjostukset.

```
3 .pro_info_text_area {  
  
    border:1px solid black;  
    background:#f2f2f2;  
    font-family:"Arial";  
    font-size:small;  
    -moz-border-radius:2px;  
    -webkit-border-radius:2px;  
    -moz-box-shadow:inset 0 0 4px #000000;  
    -webkit-box-shadow:inset 0 0 4px #000000;  
    box-shadow:inset 0 0 4px #000000;  
  
    padding:0.5em;  
}
```

KUVA 9. Tekstialueen tyylit omassa css-tiedostossaan

Web-sivustojen tyylien merkitys on laaja, kun mietitään, että Internet tavoittaa suuria määriä ihmisiä päivittäin ja useimmat yritykset mainostavat esimerkiksi omia palveluitaan siellä. Se, miten hyvin sivuston toiminnallisuudet esitetään tai kuinka selkeä rakenne sivuston sisällön esittämisessä on, voi antaa käyttäjälleen positiivisen tai negatiivisen kuvan.

2.6 Kehitysympäristö

Kehitysympäristönä opinnäytetyössä toimi Wamp Server (v. 2.1), joka sisältää Apache-palvelimen (v. 2.2.17), PHP:n (v. 5.3.5) ja MySQL-SQL-tietokantajärjestelmän (v. 5.5.8). Kehitysympäristön avulla voidaan luoda sivuja offline-tilassa, kun Apache-palvelin pyörittää sivua localhostin kautta. Wamp Serverin ohella käytössä oli Aptana Studio (v. 1.5), joka on web-pohjaisille sovelluksille tarkoitettu työkalu. Wamp Server on tarkoitettu Microsoft Windows -käyttöjärjestelmälle ja se on periaatteessa vastine Linux-käyttöjärjestelmässä käytettävälle LAMP-alustalle.

Wamp Serveriin sisältyy myös PhpMyAdmin-ohjelma, jossa voi luoda helposti uuden tietokannan, tuoda tauluja muualta tai tallentaa siirrettäväksi. PhpMyAdmin on open source -sovellus, eli sen lähdekoodi on vapaassa käytössä kaikille.

2.7 Dynaamiset ja staattiset web-sovellukset

Web-dokumentit sisältävät useasti muuttuvaa tietoa. Käyttäjän kannalta dokumentteihin tulee pystyä hakemaan muuttuvaa sisältöä esimerkiksi tietokannasta sen perusteella, mitä käyttäjä nimenomaan sillä hetkellä haluaa. (2, s. 2–4.)

Useimmissa tapauksissa koko organisaation tietojärjestelmän tiedot ovat muuttuvia. Näin ollen perinteisiin staattisiin HTML-dokumentteihin turvautuminen ei auta. Isot tietojärjestelmät vaativat myös helppokäyttöistä päivitettävyyttä. Web-dokumentteja varten on luotu tekniikoita, joilla voidaan rakentaa web-dokumentteja osittain tai kokonaan dynaamisesti. Jos tämän kaltainen dynaaminen web-sivusto on rakennettu silmällä pitäen tiettyä sovellusaluetta, voidaan puhua web-sovelluksesta. (2, s. 2–4.)

Perinteiseksi mielletty staattiset eli muuttumattomat HTML-dokumentit ovat nykyisin vähäisessä käytössä, koska niiden sisältöjen muuttaminen on työlästä ja vaatii tekijänsä toimia. Esimerkiksi monissa yrityksissä ei välttämättä ole ohjelmoijaa tai työntekijää, jolla olisi tarvittava tieto-taito tehdä muutoksia. Näin ollen on tärkeää, että tiedot ovat muutettavissa helposti. Staattiset dokumentit soveltuvat parhaiten tilanteisiin, joissa tietoja ei tarvitse päivittää tai muuttaa. (2, s. 2–4.)

2.8 Projektinhallintatyökalut

Projektinhallintaa varten on luotu suuri määrä eritasoisia ohjelmistoja. Niiden hyötynäkökulma kaiken kokoisille yrityksille on huomattava. Useimmiten yrityksillä on käynnissä monia eri projekteja ja tekijöitä sekä lukuisa määrä projekteihin liittyvää tietoa. Luonnollisesti on siis tarve pitää tiedot tallessa organisoidusti. Tarjolla käyttäjille on sekä maksullisia ohjelmistoja että

ilmaisia. Ohjelmistojen taso voi vaihdella suurestikin ja käyttäjän tulee itse arvioida, mitä ominaisuuksia hän pitää tärkeänä. (10, s. 196–199.)

Projektinhallintatyökaluja on sekä työasemakohtaisia että selainpohjaisia. Selainpohjaisissa työkaluissa etuina voidaan pitää niiden käytettävyyttä. Tämä tarkoittaa, että ohjelmistoa voidaan käyttää missä tahansa, kunhan käytössä on tietokone, Internet-yhteys ja selain. Selainpohjaisissa ohjelmistoissa tiedot ovat tallennettuina tietokantapalvelimelle ja ohjelma on yhteydessä siihen käytettäessä. Usein yrityksillä on ainakin työasemakohtaisia työkaluja, kuten Microsoft Excel -taulukkolaskentaohjelma tai muita Microsoft Officeen kuuluvia työkaluja. (10, s. 196–199.)

Työkaluista puhuttaessa voidaan mainita CASE-välineet (Computer Aided/Assisted Software Engineering), jolla tarkoitetaan kaikkia ohjelmistotyössä käytettäviä ohjelmistoja. CASE-välineitä käytetään laajasti ja niillä saavutetut edut ovat merkittäviä. Työn tuottavuus paranee, menetelmien käyttö helpottuu ja dokumentointi syntyy helposti ja nopeasti. Toisaalta yrityksille myytävät ohjelmistot saattavat olla kalliita, sillä PC-pohjaisen CASE-välineen hinta voi helposti ylittää 1000 euroa. Uusien ohjelmien valinnassa ja käyttöönotossa usein unohtuu, että tärkein osa ovat ihmiset ja prosessi, jolla ohjelmisto saadaan aikaan. Tämä on hyvä ottaa huomioon valintoja tehdessä. Yrityksen olisi hyvä myös tarkoin pohtia oma toimitapansa ennen kuin ryhtyy ostamaan ohjelmistoja. (11. s. 83–86.)

3 PROJEKTI-INFOTYÖKALUN TAULUT JA TOTEUTUS

3.1 Tietokannat

Tietokanta voi olla laaja tai suppea tietokokonaisuus, joka voi olla esimerkiksi web-sivuston menu-valikon tekstit, nettikaupan tuoteluettelo tai sitten vaikka jokin kuvatiedosto. Yleisesti ottaen voidaan miettiä, että tietokanta on loogisesti yhteenkuuluvien ja tallennettujen tietojen joukko, jota voidaan käsitellä tietokantakielellä. Tietokannat ovat laajasti käytössä maailmalla, ja sen takia tietokantasuunnittelu on tärkeää nykypäivänä. (12, s. 2–4.)

Hyötynä tietokannoissa on laajojen tietomäärien käsittely. Dynaamisessa koodauksessa tietokantojen käyttö on olennainen osa. Maailmanlaajuisesti tietokantoja käytetään muun muassa tietualan yrityksissä, pankeissa ja muissa kohteissa, joissa tarvitaan luotettavaa tapaa käsitellä esimerkiksi asiakkaiden tietoja. Tietokantoja varten on luotu erityinen ohjelmisto (tietokannan hallintajärjestelmä, DBMS). Hallintajärjestelmät ovat laajoja kokonaisuuksia ja ne tarjoavat käyttäjilleen erilaisia palveluita. Järjestelmissä on otettu huomioon tietojen muutosjoustavuus, tietoturva-asiat ja ohjelmointiin liittyvät toiminnot. Nykyään useimmiten käytetään SQL-pohjaisia relaatiotietokantoja. Relaatiotietokantojen etu vanhoihin verkkomallisiin tietokantoihin on käyttöystävällisyys. Esimerkiksi tehtäessä tietokantakyselyä voidaan keskittyä itse hakuun eikä tarvitse miettiä teknistä toteutusta. (2, s. 252–255.)

Relaatiotietokannat voivat olla saatavilla monilla eri tavoilla. Tietokantaa voidaan käyttää suoraan esimerkiksi henkilökohtaiselta työasemalta tiedostojärjestelmän tai lähiverkon tiedostopalvelimen välityksellä. Voidaan puhua asiakas-palvelinmallista, kun tietoja käsitellään tietokantapalvelimen ja asiakasohjelman välityksellä. Mallissa asiakasohjelma lähettää SQL-kyselyn tietokantapalvelimelle, joka suorittaa kyselyn ja palauttaa tulokset asiakkaalle nähtäväksi. Tämä mahdollistaa saman tietokannan käytön useiden asiakasohjelmien välillä yhtäaikaaisesti. Datana kuljetetaan ainoastaan kyselyt ja niiden tulokset. Tyypillisesti yksi tietokantapalvelin pystyy käsittelemään samanaikaisesti suuria määriä erillisiä tietokantoja. (2, s. 252–255.)

3.2 Työkalussa käytetyt taulut

Opinnäytetyön tietokantana toimi Käyttösofta Oy:n tietokanta, jonne oli erikseen luotu kolme taulua projekti-infoa varten. Ensimmäinen taulu sisältää perusviestin, joka on siis ensimmäinen tiedote, joka luodaan jo olemassa olevasta projektista. Kuvassa 10 esitellään _tunti_info-taulun rakenne. Yhteen tietueeseen kirjataan tunnistenumerot, päiväys, perusviestin otsikko, perusviesti ja projektin nimi. Nämä kaikki arvot tallentuvat tauluun, kun käyttäjä luo uuden projekti-infon.

Selaa		Rakenne		SQL		Etsi		Lisää rivi		Vienti		Tuonti		Toiminnot		Tyhjennä		Tuhoa	
Sarake		Tyyppi		Aakkosjärjestys		Attribuutit		Tyhjä		Oletusarvo		Lisätiedot							
<input type="checkbox"/>	id	int(11)						Ei		Ei mikään		AUTO_INCREMENT							
<input type="checkbox"/>	projekti_id	int(11)						Ei		Ei mikään									
<input type="checkbox"/>	projekti_nimi	varchar(50)		utf8_unicode_ci				Ei		Ei mikään									
<input type="checkbox"/>	info_otsikko	varchar(25)		utf8_unicode_ci				Ei		Ei mikään									
<input type="checkbox"/>	suljettu	int(11)						Ei		Ei mikään									
<input type="checkbox"/>	pvm	timestamp				on update CURRENT_TIMESTAMP		Ei		CURRENT_TIMESTAMP		ON UPDATE CURRENT_TIMESTAMP							
<input type="checkbox"/>	info_text	varchar(800)		latin1_swedish_ci				Ei		Ei mikään									

KUVA 10. *_tunti_info-tilin rakenne*

Käyttäjän siirtyessä projekti-infoon taulusta `_tunti_info`-taulu tuodaan perusviestin tiedot, joita voidaan kommentoida. Kommentit tallennetaan `_tunti_info_reply`-tauluun. Yhteen tietueeseen lisätään taulun, projektin, aiheen ja kommentin tunnistenumerot (`reply_id` ja `reply_id2`), käyttäjän nimi, kommentin päiväys, kommentin otsikko ja itse kommentti. Kuvassa 11 on esiteltynä `_tunti_info_reply`-taulun rakenne.

Selaa	Rakenne	SQL	Etsi	Lisää rivi	Vienti	Tuonti	Toiminnot	Tyhjennä	Tuhoa
	Sarake	Tyyppi	Aakkosjärjestys	Attribuutit	Tyhjä	Oletusarvo	Lisätiedot		
<input type="checkbox"/>	id	int(11)			Ei	Ei mikään	AUTO_INCREMENT		
<input type="checkbox"/>	projekti_id	int(11)			Ei	Ei mikään			
<input type="checkbox"/>	topic_num	int(11)			Ei	Ei mikään			
<input type="checkbox"/>	reply_id	int(11)			Ei	Ei mikään			
<input type="checkbox"/>	reply_id2	int(11)			Ei	Ei mikään			
<input type="checkbox"/>	kayttaja_nimi	varchar(50)	latin1_swedish_ci		Ei	Ei mikään			
<input type="checkbox"/>	reply_pvm	timestamp		on update CURRENT_TIMESTAMP	Ei	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP		
<input type="checkbox"/>	reply_info_otsikko	varchar(25)	utf8_unicode_ci		Ei	Ei mikään			
<input type="checkbox"/>	info_vastaus	varchar(800)	latin1_swedish_ci		Ei	Ei mikään			
<input type="checkbox"/>	checked	int(11)			Ei	Ei mikään			

Valitse kaikki / Poista valinta kaikista Valitut:

KUVA 11. tunti info reply-työkalun rakenne

Kommentit tallennetaan samaan tauluun, mutta ne jaotellaan kahden kommentitunnisteen perusteella. Perusviestin kommentit jaotellaan reply_id-sarakkeen alle ja muiden aiheiden kommentit reply_id2:n alle.

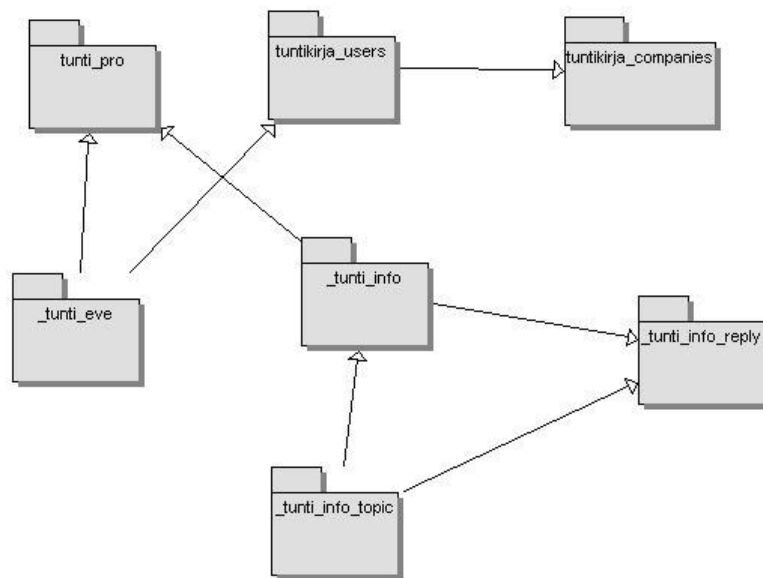
Käyttäjä voi halutessaan aloittaa uuden aiheen projekti-info-osiossa ja nämä tiedot tallennetaan _tunti_info_topic-tiluun. Taulussa yhteen uuteen aiheeseen lisätään projektin, aiheen ja taulun tunnistenumerot. Lisäksi tietueeseen kirjataan aiheen otsikko, uuden aiheen viesti, käyttäjän nimi ja päiväys. (Kuva 12.)

Sarake	Tyyppi	Aakkosjärjestys	Attribuutit	Tyhjä	Oletusarvo	Lisätiedot
<input type="checkbox"/> id	int(32)			Ei	Ei mikään	AUTO_INCREMENT
<input type="checkbox"/> pro_id	int(11)			Ei	Ei mikään	
<input type="checkbox"/> topic_num	int(11)			Ei	Ei mikään	
<input type="checkbox"/> aihe_otsikko	varchar(30)	latin1_swedish_ci		Ei	Ei mikään	
<input type="checkbox"/> aihe_viesti	varchar(800)	utf8_unicode_ci		Ei	Ei mikään	
<input type="checkbox"/> aihe_kayttaja	varchar(50)	latin1_swedish_ci		Ei	Ei mikään	
<input type="checkbox"/> aihe_pvm	timestamp		on update CURRENT_TIMESTAMP	Ei	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP

KUVA 12. Taulun _tunti_info_topic rakenne

Aiheen otsikko, viesti ja käyttäjä tiedot tallennetaan varchar-tyyppiseksi muuttujaksi ja tunnistenumerot (id:t) integer-tyyppiseksi. Aiheen viestille, otsikolle ja käyttäjän tiedoille on annettu maksimimerkkimäärä, jonka käyttäjä voi syöttää. Itsestään koon määrittäminen tietokantaan ei riitä, sillä jos esimerkiksi syötetään tekstikenttään yli sallittu määrä merkkejä, tietokantaan tallentuu pelkästään aiemmin määritelty määrä merkkejä. Ohjelmaan voitaisiin lisätä tekstikenttään toiminnallinen koodi, joka estää käyttäjää syöttämästä liikaa merkkejä.

Kuvassa 13 on esitelty taulut, jotka ovat käytössä koko ohjelmassa. Tämä työ keskittyy käsittelemään ainoastaan projekti-info-osiota. Pääasiassa osiossa käytössä on _tunti_info- ja _tunti_info_topic-tilut, jotka ovat yhteydessä _tunti_info_reply-tiluun.



KUVA 13. Projekti-infotyökalussa käytetyt taulut

4 PROJEKTI-INFOT

Opinnäytetyön aiheena oli luoda dynaaminen viestittämistoiminnallisuus Käyttösofta Oy:lle. Työkalun avulla käyttäjät voivat luoda uuden viestittämisosion eli projekti-info-osion. Osio luodaan jo aiemmin luodusta projektista, jonka jälkeen sitä voidaan käyttää. Työkalu koostuu muutamasta muustakin toiminnosta, mutta tämä työ keskittyy käsittelemään viestittämisosiota.

Työ tehtiin 18.5.2011–30.8.2011 Kempeleessä Käyttösofta Oy:n tiloissa. Työn alkuvaiheessa luotiin tavoitteet ja suunnitelma, jota noudatettiin läpi opinnäytetyön. Läpi työn pidettiin palavereja, joissa pyrittiin selvittämään eteneminen ja luotavat muutokset.

4.1 Työn vaatimukset

Työn päävaatimuksina oli luoda helppokäyttöinen ja dynaaminen viestienlähetystyökalu. Työkalu muistuttaa foorumityyppistä toteutusta, missä käyttäjä näkee, kuka on lähettänyt viestin, milloin viesti on lähetetty ja mitä viesti sisältää. Kaikki viestit tallennetaan tietokantaan.

Sovelluksen dynaaminen toimivuus on tärkeä ominaisuus, koska työssä infotiedotteiden määrää alustavasti on mahdoton määrittää. Viestittämisosiossa määriteltiin toimintoina mm. kommenttien luonti ja muokkaus. Tarkemmat kuvaukset toiminnoista on kuvailtu luvussa 5.2. Tyyli vaatimuksina oli tilaajan puolelta esityksen selkeys ja tiiviys.

4.2 Sovelluksen käyttö

Työn alkuvaiheessa paneuduttiin luomaan Luo projekti-info -toiminnallisuutta, jossa käyttäjä luo perusotsikon ja perusviestin jo aiemmin luotuun projektiin. Käyttäjän syöttämät arvot tallennetaan tietokantaan ja projekti-info on luotu. (Kuva 14.)

[Takaisin etusivulle](#)

Projektin nimi:	Muokattu projekti
Otsikko:	
Alku-info:	
	Tallenna

KUVA 14. Luo uusi projekti-info -toiminto

Tallennuksen jälkeen käyttäjä ohjautuu takaisin pääsivulle, jossa projekti-infot ovat heti käytettävissä. Käyttäjä valitsee seuraavaksi haluamansa projekti-infon ja ohjautuu linkin kautta käyttämään sitä.

Projekti-info-osiossa käyttäjä voi kommentoida perusviestiä, muokata sitä tai luoda uuden projektiin liittyvän aiheen. Kommentteja käyttäjä voi muokata, poistaa tai merkata tehdyksi. Useimmiten kommentit voidaan ajatella tehtäviksi, joita projektissa käsitellään, joten työssä on mahdollista merkata kommentti tehdyksi. Syötettäessä kommenttia käyttäjää vaaditaan täyttämään kommentin otsikko ja itse kommentti. Käyttäjä lähettää tiedot eteenpäin painaessaan Lähetä-nappia, jolloin tiedot lähetetään tietokantaan. (Kuva 15.)

Otsikko:	Käyttäjä:	Päivämäärä:
muokattu alkuinfo	Juho Suhonen	2011-07-04 15:25:32

Muokattu Projektin tavaraaMuokattu Projektin tavaraaMuokattu Projektin tavaraa # Tähän lisäys vielä

1 kommentti otsikko	Juho Suhonen	2011-08-25 11:59:14
1 kommentin viesti aiheeseen 1		
2 otsikko	Juho Suhonen	2011-08-25 11:59:44
3 otsikko	Juho Suhonen	2011-08-25 11:59:53

Kirjoittaja:
Juho Suhonen

Otsikko:

Kirjoita kommenttisi:

Lähetä

KUVA 15. Kommentin lisäys

Käyttäjän ohjautuessa takaisin projekti-info sivulle uusi kommentti on nähtävillä. Käyttäjä voi halutessaan lisätä uuden keskusteluaiheen sivulle. Lisättäessä uutta keskusteluaihetta tulee käyttäjän täyttää uuden aiheen otsikkoviesti ja alkuviesti. Aiheet erottuvat harmaalla taustavärillä kommentteista ja niitä varten on tehty toiminto, jolla voi piilottaa tai näyttää kaikki aiheeseen liittyvät viestit.

Projekti-infossa alkuviestit ja kommentit sisältävät viestin (tai kommentin), viestin otsikon, päiväyksen ja käyttäjänimen. Jos käyttäjä haluaa muokata toisen kommenttia, tallennettaessa käyttäjänimi muuttuu kyseisen käyttäjän mukaisesti. Toiminnot käsitellään luvussa 5.2 tarkemmin.

Projekti-infon käyttö on helppoa ja sen etuna on käytettävyys. Infotiedoteosiota voidaan käyttää missä tahansa, kunhan on saatavilla tietokone, Internet-yhteys ja selain.

4.3 Toiminnot

Luo uusi projekti-info. Toiminto, jossa luodaan uusi projekti-info. Käyttäjää pyydetään valitsemaan projekti, jota varten info luodaan, perusviestin otsikko ja itse perusviesti. Lähetettäessä tiedot tallennetaan Käyttösofta Oy:n tietokantaan tauluun _tunti_info. Infotiedotteet ilmestyvät päävalikkoon, josta valitaan haluttu projekti-info. Käyttäjä voi kommentoida

infotiedotteita tai muokata niitä. Näkymästä voi nähdä, kuka on kirjoittanut kommentin ja milloin se on kirjoitettu. Siirryttäessä projekti-infoon luotu perusviesti näkyy sivulla ylimmäisenä.

Lisää kommentti -toiminnolla käyttäjä voi lisätä kommentteja eri aihealueisiin. Käyttäjän tulee täyttää lomakkeeseen kommenttinsa otsikko ja itse kommentti. Tallennettaessa tiedot lähetetään tauluun _tunti_info_reply. Yksi tietue taulussa sisältää kommentin tunnisteet (taulun tunnisteiden, projektin tunnisteiden, kommentin tunnisteet sekä aiheen tunnisteiden), kommentin otsikkoviestin, kommentin, kommentin lähettäjän ja päiväyksen. Kommentin tallennusvaiheessa tietokantaan tallentuu heksadesimaalikoodina punainen väri. Väriä käytetään määrittämään, onko kommentti uusi ja niin sanotusti tekemätön. Kommentteja tässä työssä voidaan ajatella tehtäväksi, joten aluksi ne merkataan tekemättömiksi. Tauluun tallennetaan värin ohella tarkastusarvo kommentteille. 0 vastaa tekemätöntä tehtävää ja 1 tehtyä. (Kuva 19.)

Lisää uusi aihe -toiminnolla käyttäjä voi luoda uuden ala-aihealueen, jossa luodaan alkuviesti uudelle ala-aihealueelle. Tämän jälkeen käyttäjät voivat kommentoida samaan tapaan uutta ala-aihealuetta. Uusi aihealue tallentuu tietokantaan tauluun _tunti_info_topic. Tauluun tallennetaan tunnistetiedot, uuden aiheen otsikkoviesti, aiheviesti, päiväys ja käyttäjänimi.

Muokkaa-toimintoa voidaan käyttää jokaisessa kommentissa ja alkuviestissä. Muokkaa-toiminnossa käyttäjä ohjataan muokkaamaan halutun viestin otsikkoviesti ja itse viesti. (Kuva 16.)

VERSO

Easy Software for CarSales

Vaihda salasana / käyttäjätunnus

Käyttäjä: **Juho Suhonen**

Kirjaudu ulos

Takaisin

Otsikko:	Kirjoittaja:	Päiväys:
1 kommentti aihe 2	Juho Suhonen	2011-09-05 08:16:25
<div>testiää testiää testiää testiää testiää testiää testiää testiäätestiää</div>		
<div>Tallenna</div>		

KUVA 16. Kommentin muokkaus

Uudet tiedot päivitetään tietokantaan, jonka jälkeen käyttäjä ohjautuu takaisin projekti-infosivulle. Koodissa muokkausta varten on luotu useampi funktio muokkaa-toiminnolle. Käyttäjän ohjautuessa muokkuspainikkeen kautta muokkaussivulle välitetään index.php-tiedostoon projektin tunnistenumero. Kommenttien muokkauksessa välitetään lisäksi myös aiheen tunnistenumero ja kommentin tunnistenumero. Nämä tiedot välitetään forms_function.php-tiedostoon, jossa funktio EditProjectInfos avaa valitun perusviestin tiedot hakemalla tiedot tietokannasta. Funktio avaa uuden formin eli lomakkeen, johon käyttäjä syöttää tiedot. Käyttäjän tulee syöttää ainoastaan aiempi otsikkoviesti ja itse viesti. Käyttäjänimi ja päiväys tallentuvat automaattisesti, kun muutetut tiedot tallennetaan. Kuvassa 17 on kuvattuna tietokantahaku funktiossa EditProjectInfos.

```
4 function EditProjectInfos($link, $prefix, $company, $p_info_id )
5 {
6     $sql = "SELECT * FROM " . $prefix . " _tunti_info WHERE projekti_id = '" . $p_info_id . "'";
7     $result = mysql_query($sql, $link);
8
9
10
11     $projektinimi = mysql_result($result, 0, "projekti_nimi");
12     $info_otsikko = mysql_result($result, 0, "info_otsikko");
13     $viesti = mysql_result($result, 0, "info_text");
14     $etu_k = $_SESSION["fname"]; // sessionista etu ja sukunimi
15     $suku_k = $_SESSION["lname"];
16     $pvm = mysql_result($result, 0, "pvm"); //pvm
17 }
```

KUVA 17. Tietokannasta haetaan muokattavan alkuviestin tiedot

Kun uudet tiedot lähetetään, ohjelma siirtyy index.php-tiedostoon, josta se ohjataan takaisin form_functions.php-tiedostoon tallennusfunktioon. Tallennusfunktiossa tallennetaan muuttujiin formiin syötetyt arvot ja päivitetään ne tietokantaan. (Kuva 18.)

```

function SQLSaveEditedProjectInfo ($link, $prefix, $p_info_id )
{
    $pro_id = $_POST["pro_info_id"];
    $kayttaja = $_POST["kayttaja_1"];
    $pro_i_otsikko = $_POST["edit_i_header"];
    $muokattu_viesti = $_POST["edit_pro_reply_text_area"];

    $sql = "UPDATE " . $prefix . "_tunti_info SET projekti_id = '" . $pro_id . "' WHERE projekti_id = '" . $p_info_id . "'";
    $result = mysql_query($sql, $link) or die("<span class='notify_text'>Virhe: Projektin nimen tallentaminen epäonnistui.</span><br />");

    $sql = "UPDATE " . $prefix . "_tunti_info SET kayttaja = '" . $kayttaja . "' WHERE projekti_id = '" . $p_info_id . "'";
    $result = mysql_query($sql, $link) or die("<span class='notify_text'>Virhe: Projektin nimen tallentaminen epäonnistui.</span><br />");

    $sql = "UPDATE " . $prefix . "_tunti_info SET info_otsikko = '" . $pro_i_otsikko . "' WHERE projekti_id = '" . $p_info_id . "'";
    $result = mysql_query($sql, $link) or die("<span class='notify_text'>Virhe: Projektin nimen tallentaminen epäonnistui.</span><br />");

    $sql = "UPDATE " . $prefix . "_tunti_info SET info_text = '" . $muokattu_viesti . "' WHERE projekti_id = '" . $p_info_id . "'";
    $result = mysql_query($sql, $link) or die("<span class='notify_text'>Virhe: Projektin nimen tallentaminen epäonnistui.</span><br />");
}

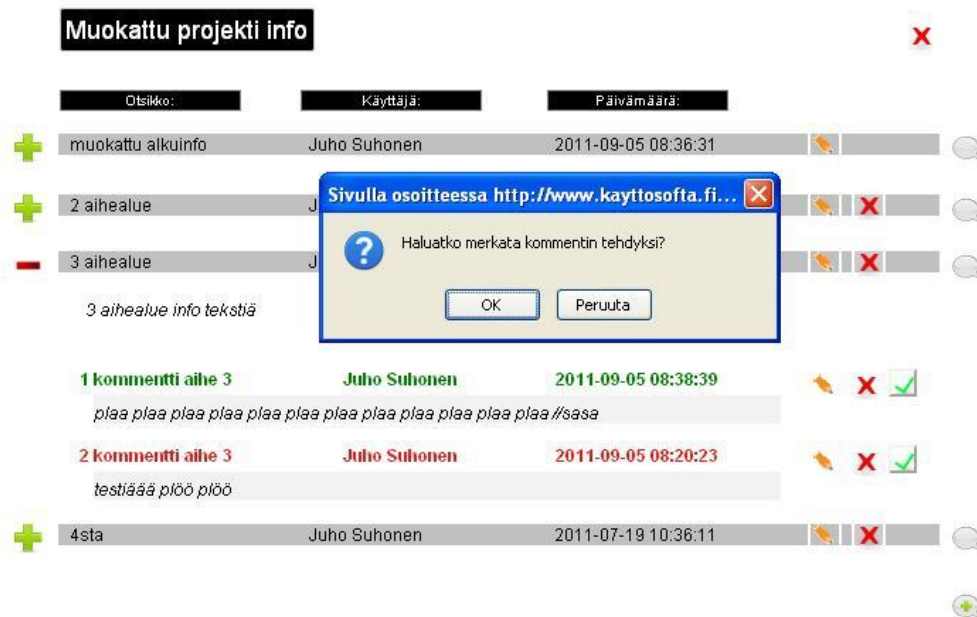
```

KUVA 18. Tallennusfunktiossa lähetetään päivityskomento tietokantaan

POISTA-toiminnolla voidaan tuhota kommentteja, keskusteluaiheita tai koko projekti-info. Valitessaan POISTA-toiminnon käyttäjä ohjautuu varmistussivulle, josta voi joko hyväksyä poiston tai perua sen. Koodissa siirrytään lopulta funktioon, jossa tuhoetaan tiedot taulusta. Kun kyseessä on koko projekti-infon poisto, tuhoetaan kaikki siihen liittyvät kommentit ja viestit.

MERKKAUS-toiminnolla voidaan merkata kommentti tehdyksi. Kommentin luontivaiheessa tietokantaan _tunti_info_reply-tauluun (joka on kommentteja varten) tallennetaan punainen fonttiväri. Merkkauksnapista painettaessa käyttäjälle ilmestyy varmistuslaatikko, jossa kysytään, merkataanko tehtävä tehdyksi. (Kuva 19.)

Varmistuksen jälkeen käyttäjä ohjautuu takaisin projekti-infosivulle ja kommentin fonttiväri on muuttunut vihreäksi. Koodissa varmistuslaatikko on luotu käyttämällä JavaScript-ohjelmointikieltä. Käyttäjän varmentaessa muutoksen tietokantaan lähetetään uuden värin heksadesimaalikoodi ja arvo 1 checked-nimiseen sarakkeeseen.



KUVA 19. Kommentin merkkkaus

Kommenttirivillä fonttiväri haetaan tietokannasta jo luontihetkestä lähtien, joten uusi väri ilmestyy saman tien merkkauksen jälkeen. Kuvassa 20 on värin muunnoksen toteutus. Periaatteessa tietokannassa oleva värin heksadesimaalikoodi haetaan ja tallennetaan muuttujaan color, jota kutsutaan kommenttirivillä.

```
for ($j=0; $j<$num2; $j++)
{
    $kayttaja_nimi = mysql_result($result2, "$j", "kayttaja_nimi"); //käyttaja nimi
    $reply_pvm = mysql_result($result2, "$j", "reply_pvm"); // kommentin aika
    $reply_otsikko = mysql_result($result2, "$j", "reply_info_otsikko"); // kommentin otsikko
    $info_vastaus = mysql_result($result2, "$j", "info_vastaus"); //kommentti
    $color = mysql_result($result2, "$j", "color"); //väri-koodi

    echo "
        <div class=\"info_switch_div$reply_count\">
            <div class=\"collapsible\">

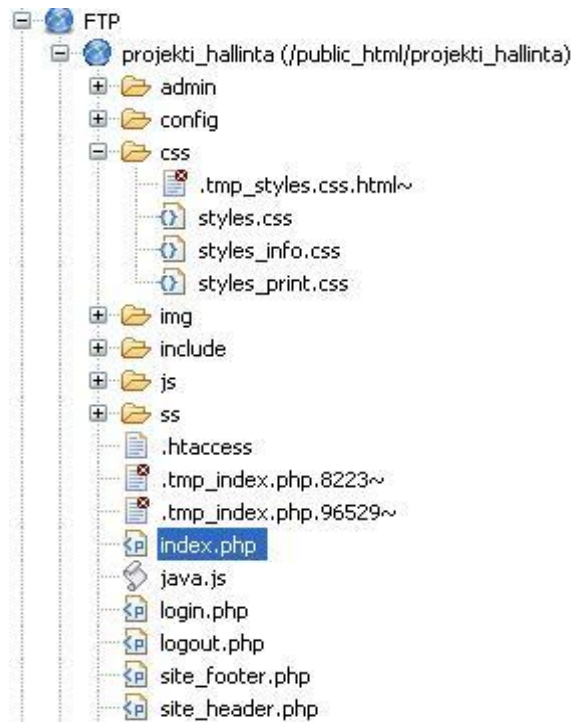
                <a class=\"collapse\" id=\"show_reply\" style=\"color:$color;\" href=\"\">
```

KUVA 20. Color-muuttuja koodissa

4.4 Rakenne

Projekti-infon rakenne on kaksiosainen, koska perusviestit on eritelty muista infoista. Tämä tarkoittaa, että koodissa on siis erikseen funktiot perusviestin toiminnoille kuten myös muille aiheille. Perusviestin tiedoille ja ala-aihealueiden tiedoille on olemassa omat taulut. Ne ovat

yhteydessä kommentti-tauluun, johon sekä perusviestin kommentit että muiden aiheiden kommentit ovat tallennettu. Koodissa kommenttien tiedot haetaan tunnistenumeroiden perusteella. Projekti-infolle, aiheelle ja kommentille on olemassa omat tunnistenumerot. Projekti-infon tiedostorakenne työssä koostuu kahdesta tyylitiedostosta (.css), muutamasta rakennetiedostosta ja funktiotiedostosta (.php). Muita tiedostoja ovat myös toiminnallisuutta varten JavaScript-ohjelmointikielellä tehdyt tiedostot ja tietokantayhteys-tiedostot. (Kuva 21.)

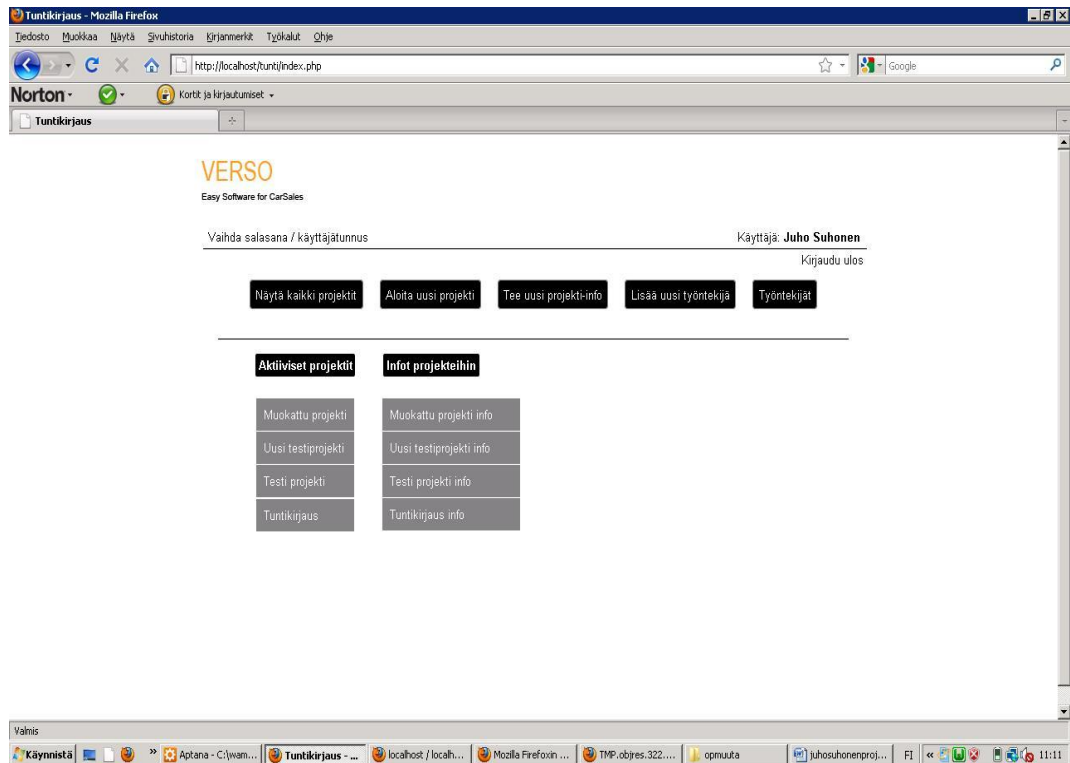


KUVA 21. Tiedostorakenne

4.5 Käyttöliittymä

Käyttöliittymä on yksinkertainen ja tyylit ovat yhtenevät läpi sivuston. Perusväritys on musta-valko-harmaa. Päävalikossa linkeille haetaan tyylit css-tiedoston tyyliohjeiden mukaisesti.

(Kuva 22).



KUVA 22. Päävalikon perusnäkö 14.6.2011

Päävalikossa käyttäjä voi valita haluamansa toiminnot. Päälinkit sijaitsevat keskellä sivua ja luodut projektit tai projekti-infot löytyvät niiden alta. Kirjautuneen käyttäjän nimi on esillä päävalikossa oikealla ylhäällä. (Kuva 23.)



KUVA 23. Päävalikon perusnäkö 30.8.2011

Työkalussa on luotu useimmille toiminnoille omat tyyliohjeet, kuten esimerkiksi linkeille sekä yleisiä ohjeita että yksilöllisiä ohjeita. Yksilöllisillä ohjeilla tarkoitetaan yhden tai parin linkin tyylin luomista eri tapaan kuin muut, eli niille on luotu omat ohjeet. Linkkien korostaminen ja erottaminen voi antaa näyttävämmän tuloksen. Työssä esimerkiksi eri toimintojen linkit erottuvat muista kuvilla. Linkeille on myös normaalityylien lisäksi luotu toiminnallisia tyyliä. Esimerkiksi käyttäjän viedessä kursorin linkin päälle voi linkin reunaväri tai taustaväri muuttua.

Kuvassa 24 on esiteltynä projekti-infon perusnäkö. Otsikkona näkyy valitun projekti-infon nimi. Plusmerkeistä käyttäjä voi avata näkyville koko aiheen eli kaikki siihen liittyvät viestit. Aiheen ollessa avattuna plusmerkin tilalle ilmestyy miinusmerkki, josta aihe voidaan sulkea. Kommentit voidaan samaan tapaan aukaista ja sulkea.

VERSO
Easy Software for CarSales

Vaihda salasana / käyttäjätunnus Käyttäjä: **Juho Suhonen** Kirjautu ulos

[Takaisin etusivulle](#)

Muokattu projekti info

Otsikko:

Käyttäjä:

Päivämäärä:

	muokattu alkuinfo	Juho Suhonen	2011-09-05 08:36:31			
	2 aihealue	Juho Suhonen	2011-09-05 08:38:14			
	3 aihealue	Juho Suhonen	2011-07-04 15:44:55			

3 aihealue info tekstiä

1 kommentti aihe 3	Juho Suhonen	2011-09-05 08:38:39				
plaa plaa plaa plaa plaa plaa plaa plaa plaa plaa plaa plaa //sasa						
2 kommentti aihe 3	Juho Suhonen	2011-09-05 08:20:23				
testiäää plöö plöö						
	4sta	Juho Suhonen	2011-07-19 10:36:11			

KUVA 24. Info-osio perusnäkö 31.8.2011

Aiherivi toimii linkkinä, joten sitä painamalla voi nähdä aiheen alkuviestin. Aiheet on eroteltu kommentteista harmaalla taustavärillä. Itse perusviestit ja kommentit ovat kursivoituna. Toiminnallisilla linkeillä on yksilölliset kuvat, jotka kuvastavat niiden aiheita.

Kynän kuva on MUOKKAUS-toimintoa varten. Punainen rastimerkki merkitsee POISTA-toimintoa ja vihreä oikeinmerkki kommentin merkkausta. Puhekuplan kuva on Luo uusi kommentti -toimintoa varten ja puhekuplan kuva, jossa on keskellä plusmerkki, on tunnus Luo uusi aihealue -toimintoa varten. (Kuva 24.)

VERSIO
Easy Software for CarSales

Vaihda salasana / käyttäjätunnus Käyttäjä: Juho Suhonen
Kirjaudu ulos

Takaisin etusivulle

Muokattu projekti info

Otsikko:	Käyttäjä:	Päivämäärä:
muokattu alkuinfo	Juho Suhonen	2011-09-05 08:36:31

Kirjoittaja:
Juho Suhonen
Otsikko:
Kirjoita kommenttisi:
Lähetä

KUVA 25. Kommentin lisäyslaatikon yksilölliset tyyliohjeet

Linkkien lisäksi kommenttilaatikoilla on yksilöllisiä tyyliohjeita. Esimerkiksi tekstikentille on annettu tumma varjo ja reunoja on pyöristetty. Myös käyttäjän syöttämälle fontille on annettu omat tyyliohjeet. (Kuva 25.)

Käyttöliittymän vaatimuksina Käyttösofta Oy:n puolelta oli esitystapa, joka olisi selkeä ja tiivis. Työhön tehty toteutus vastaa näitä vaatimuksia, sillä se pitää projekti-infon suppeana ja käyttäjälle miellyttävänä. Osion peruskäytössä on todennäköistä, että yhteen projekti-infoon kirjataan lukuisia aiheita ja kommentteja. Näin ollen on tärkeää, että aiheet voidaan piilottaa ja näyttää tarvittaessa. Myös tiivis esitystapa helpottaa aiheiden hahmottamista ja käyttäjä löytää niiden kommentit paremmin.

5 YHTEENVETO

Opinnäytetyössä rakennettiin projektihallintatyökaluun viestienlähetystoiminnallisuus Käyttösofta Oy:lle. Tavoitteena oli luoda dynaaminen ja käyttäjäystävällinen projekti-info yrityksen käyttöön, päivittäisten projektien hoitoon. Työ aloitettiin 18.5.2011 ja sitä tehtiin yhtä jatkoisesti läpi kesän, aina 30.8.2011 asti Käyttösofta Oy:n tiloissa. Projektin päättyessä tuloksena saatiin aikaan toiminnallisuus, joka kattaa perustoiminnot, kuten projekti-infon luonnin, viestien kommentoinnin ja muokkauksen. Infoja tai kommentteja voidaan myös poistaa.

Projekti-info sisältää tärkeimmät ominaisuudet, jotka vaatimuksissa oli määritelty. Työn olisi voinut toteuttaa monella tavalla, mutta mielestäni luodut ratkaisut ovat toimivia käyttötarkoitusta varten. Esimerkkinä jos projekti-info luodaan jo valmiista projektista, se yksilöi projekti-infon vastaamaan tiettyä projektia. Luo projekti-info -toiminto toimii työssä pohjana, josta tuodaan viestittämisosioon alkuviesti. Perusviestit ja niiden kommentit on eroteltu muista viesteistä siten, että niille on luotu omat funktiot. Tämän kaltaiseen ratkaisuun päädyin, sillä kun käyttäjä luo projekti-infon, sen alkuviesti tallennetaan eri tauluun kuin muiden ala-aiheiden viestit. Info-osiossa ei myöskään ole vaihtoehtoa tuhota alkuviestejä. Alkuviestit voidaan tuhota ainoastaan tuhoamalla koko projekti-info. Tämä tarkoittaa sitä, että ensisijaisesti aluksi luotu alkuviesti on pohja projekti-infolle ja sitä ei ole tarve erikseen poistaa.

Viestittämisosio on dynaaminen, joten siihen voidaan luoda käytännössä lukematon määrä uusia kommentteja tai ala-aihealueita. Mielestäni tämä on kenties tärkein ominaisuus koko työkaluosiossa, sillä ilman tätä toteutustapaa olisi raskas toteuttaa vaadittuja toimintoja.

Työ onnistui mielestäni hyvin ja tulevaisuudessa sitä olisi helppo kehittää. Työhön voisi lisätä esimerkiksi tarkemmin oikeudet käyttäjille tai tyylejä voitaisiin muokata näyttävämmäksi. Työn valmistumisvaiheessa ei ole tietoa osion jatkokehittämisestä, kun osion käyttäjäryhmänä toimii ainoastaan yrityksen väki.

LÄHDELUETTELO

1. PHP. 2011. Saatavissa: <http://fi.wikipedia.org/wiki/PHP>. Hakupäivä 10.7.2011.
2. Rantala, Ari 2005. Web-ohjelmointi. Porvoo: Docendo Finland Oy.
3. PHP. 2011. Saatavissa: http://www.ohjelmointiputka.net/oppaat/opas.php?tunnus=php_01. Hakupäivä 9.7.2011.
4. MySQL. 2011. Saatavissa: <http://fi.wikipedia.org/wiki/Mysql>. Hakupäivä 11.7.2011.
5. Reese, George – Yarger, Randy Jay – King, Tim – Williams, Hugh E. 2002. Managing & using MySQL. 2. painos. 1005 Gravenstein Highway North, Sebastopol: O'Reilly & Associates Inc.
6. HTML. 2011. Saatavissa: <http://fi.wikipedia.org/wiki/Html>. Hakupäivä 11.7.2011.
7. Musciano, Chuck – Kennedy, Bill 2002. HTML & XHTML the definitive guide. 5. painos. 1005 Gravenstein Highway North, Sebastopol: O'Reilly & Associates Inc.
8. Peltomäki, Juha 2011. JavaScript. Jyväskylä: Docendo Finland Oy.
9. Räsänen, Seppo 2000. CSS-tyylit. Saatavilla: <http://webd.savonia.fi/home/ktrasse/muut/haka/css.pdf>. Hakupäivä 15.8.2011.
10. Lehtimäki, Timo 2006. Ohjelmistoprojektit käytännössä. Jyväskylä: Readme.fi.
11. Haikala, Ilkka – Märijärvi, Jukka 2002. Ohjelmistotuotanto. 8. painos, Pieksämäki: Talentum Media Oy.

12. Hovi, Ari – Huotari, Jouni – Lahdenmäki, Tapio 2003. Tietokantojen suunnittelu & indeksointi. Porvoo: Docendo Finland Oy.